



مهندسی نرم افزار ۱

مدرس: رامتین خسروی

بهار ۱۴۰۳

تکلیف صفر

آشنایی با جاوا

هدف این تکلیف مرور ویژگی‌هایی از زبان جاوا است که ممکن است کمتر با آن آشنا باشید اما در جریان انجام تکلیف‌های بعدی به آنها نیاز پیدا می‌کنید. این تکلیف در چند گام تعریف شده و در هر گام به بخش‌هایی از مراجع آموزش جاوا ارجاع داده شده است که مطالعه آنها توصیه می‌شود. در خیلی از موارد به بخشی‌هایی از کتاب Core Java for the Impatient ارجاع داده شده که با کته‌نوشت ال C به آن اشاره شده است.

گام ۱ - نصب جاوا

در صورتی که روی سیستم خود ابزارهای توسعه جاوا (JDK) نصب نکرده‌اید این کار را انجام دهید. برای این که مشکلات تحریم دردرس کمتری ایجاد کند توصیه می‌شود از [openjdk](#) استفاده کنید که نصب آن با apt قابل انجام است. برای انجام پروژه‌های درس توصیه می‌شود نسخه ۱۷ یا بالاتر را نصب کنید.

گام ۲ - ایجاد پروژه

در این گام یک پروژه ایجاد می‌کنیم که در گام‌های بعد آن را کامل می‌کنیم. از آنجا که در گام‌های بعد از کتابخانه CSVReader استفاده خواهیم کرد، این پروژه باید با ابزار میون (Maven) ساخته شود. برای ایجاد پروژه می‌توانید [مستقیماً از میون استفاده کنید](#) یا برای راحتی کار می‌توانید از [محیط برنامه‌نویسی آیدیا](#) (IntelliJ IDEA) استفاده کنید. نسخه رایگان Community Edition این ابزار برای انجام پروژه‌های این درس کافی است. پس از نصب آیدیا، یک پروژه جدید ایجاد کنید و Build System آن را Maven تعیین کنید. نیازی

به ایجاد مخزن گیت برای پروژه نیست و برای راحتی کار می‌توانید Add sample code را تیک بزنید تا کد نمونه برایتان تولید شود.

گام ۳ - مطالعه کلاس Date

در پروژه ساخته شده، یک کلاس Date ایجاد کنید و بدنه آن را از روی [پیاده‌سازی داده‌شده](#) کپی کنید. این پیاده‌سازی را مطالعه کنید و مطمئن شوید نکات زیر را می‌دانید:

- اشیاء در جاوا همیشه از طریق ارجاع (مشابه اشاره‌گر در C++) مورد استفاده قرار می‌گیرند [CJ - Section 2.1.2]
- چرا فیلدهای این کلاس به صورت final تعریف شده‌اند [CJ - Section 2.3.6]
- سازوکار استثناءها در جاوا چگونه است [CJ - Section 5.1]
- کلاس Object در کتابخانه استاندارد چیست؟ سه متد equals و hashCode و toString چه نقشی دارند؟ [CJ - Section 4.3]

بدنه متد main کلاس Main را تغییر دهید: یک شیء تاریخ بسازید که معادل ۱۴۰۲/۱۱/۱۸ باشد. فردای این روز را در خروجی بنویسید.

گام ۴ - اضافه کردن مقایسه به کلاس Date

ابتدا درباره رابط‌ها (interfaces) در جاوا مطالعه کنید. برای یک شروع نسبتاً ساده می‌توانید از [این خودآموز](#) استفاده کنید. [CJ - Section 3.1] هم منبع خوبی برای مطالعه است. درباره رابط Comparable مطالعه نمایید [CJ - Section 3.3] و کلاس Date را Comparable نمایید.

گام ۵ - اضافه کردن کلاس‌های Membership و Player

ابتدا یک کلاس به نام Membership تعریف کنید که مدل‌کننده عضویت در یک تیم در یک بازه زمانی است. برای آن سه عضو تعریف کنید: نام تیم از نوع رشته، تاریخ شروع و پایان عضویت از نوع تاریخ (کلاس Date). کنترل‌های لازمه را در سازنده انجام دهید و در صورتی که آرگومان‌های آن نامعتبر باشند IllegalArgumentException پرتاب کنید. همچنین متدی تعریف کنید که یک بازه‌ی دیگر را می‌گیرد و بسته به این که دو بازه هم‌پوشانی (حتی در یک روز) داشته باشند مقدار درست یا نادرست برگرداند. دو بازه $[b1, e1]$ و $[b2, e2]$ هم‌پوشانی دارند اگر $(e1 \geq b2) \&\& (b1 \leq e2)$. متد دیگری برای این کلاس تعریف کنید تا تعداد روزهای یک عضویت را برمی‌گرداند.

سپس یک کلاس به نام Player تعریف کنید که علاوه بر نام بازیکن، تاریخچه عضویت او در تیم‌های مختلف را در قالب یک لیست نگه می‌دارد. برای تعریف این لیست از ArrayList استفاده کنید که شبیه به vector در

C++ است. یک متد برای Player تعریف کنید که بتوان با آن یک سابقه عضویت به آن بازیکن اضافه کرد. برای این متد عدم هم‌پوشانی عضویت جدید با عضویت‌های دیگر بازیکن را کنترل کنید.

گام ۶ - محاسبات تابعی

متدی برای بازیکن تعریف کنید که نام یک تیم را می‌گیرد و تعداد کل روزهایی را که بازیکن عضو آن تیم بوده را برمی‌گرداند. برای پیاده‌سازی این متد از سبک محاسبات تابعی (functional) استفاده کنید. به این منظور با امکانات stream در جاوا آشنا شوید ([خودآموز مقدماتی](#)، [CJl - Sections 8.1-8.3]) و از عبارتهای لامبدا [CJl - Section 3.4] استفاده کنید. به این ترتیب بدنه این متد در یک خط قابل پیاده‌سازی است.

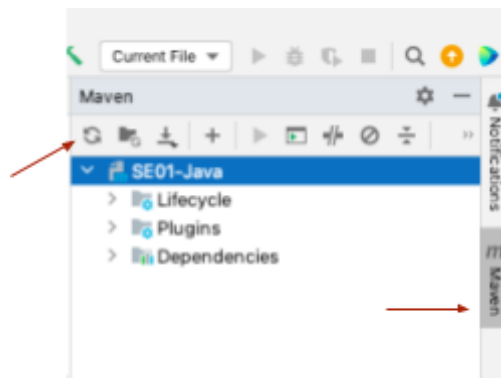
گام ۷ - خواندن سوابق از فایل

در این گام متد main برنامه را کامل می‌کنیم تا اطلاعات سوابق بازیکنان را از یک فایل سی‌اس‌وی بخواند و محاسباتی را روی آنها انجام دهد. به این منظور، از کتابخانه [OpenCSV](#) است استفاده می‌کنیم. پیشنهاد می‌شود برای انجام این گام از [CSVReader](#) استفاده کنید.

برای اضافه کردن کتابخانه OpenCSV به پروژه، لازم است pom.xml را که ساختار پروژه شما و وابستگی‌های آن به کتابخانه‌های خارجی را برای میون توصیف می‌کند تغییر دهیم. به این منظور این فایل را در پروژه خود پیدا کنید (در همان پوشه پروژه) و آن را در آیدیا باز کنید. در آخرین قسمت عنصر project (بعد از properties) باید بخش زیر را کپی نمایید:

```
<dependencies>
  <dependency>
    <groupId>com.opencsv</groupId>
    <artifactId>opencsv</artifactId>
    <version>5.8</version>
  </dependency>
</dependencies>
```

(اگر احیاناً pom.xml شما از قبل عنصر dependencies دارد، فقط dependency فوق را داخل dependencies موجود کپی کنید). ممکن است بعد از انجام این تغییر شماره نسخه (۵.۸) به رنگ قرمز در بیاید. در این صورت با استفاده از امکان Reload All Maven Projects (شکل زیر) این مشکل را حل کنید.



پس از اضافه کردن وابستگی فوق در pom.xml می‌توانید از کلاس‌های کتابخانه مذکور (مثل CSVReader) استفاده کنید. برای حل این قسمت می‌توانید کد خود را برای خوانایی بیشتر در قالب متدهایی استاتیک در کلاس Main سازماندهی کنید.

فرض کنید اطلاعات سوابق بازیکنان در یک فایل به نام input.csv در پوشه پروژه شما موجود است. محتوای این فایل می‌تواند شبیه به این باشد:

```
Gholam,Golgozar,1,1,1380,10,5,1381
Gholam,Esteghlal,15,2,1382,14,2,1385
Gholam,Golgozar,1,4,1385,31,5,1390
Xi,Golgozar,1,1,1381,1,5,1388
Hooshang,Esteghlal,1,1,1379,29,12,1390
Hooshang,Golgozar,1,1,1391,29,12,1401
```

در نهایت، با استفاده از مثال فوق، محاسبه کنید غلام چند روز عضو تیم گل‌گهر بوده است.

نحوه تحویل

در پاسخ تکلیف پوشه پروژه خود را (به جز زیرپوشه target) در قالب یک فایل زیپ به نام A0-{StudentNo}.zip فشرده کنید و در محل مشخص‌شده در ای‌لرن بارگذاری نمایید (به جای {StudentNo} شماره دانشجویی خود را قرار دهید).